

REMARKS

Status of Claims

The Office Action mailed January 10, 2008 was received and reviewed. Each of claims 1-4 and 13-16 stand rejected. Claims 1, 4, 13-14, and 16 are currently amended, and claims 3 and 15 are canceled. Reconsideration of the above-identified application in view of the following remarks is respectfully requested.

Summary of Interview

Applicants thank Examiner Wai for the courtesy of a telephone interview with Applicants' representatives on April 1, 2008. During the interview, the § 101 rejections of claims 13-16 were discussed. Applicants proposed to amend these claims to read "[o]ne or more computer-storage media having computer-executable instructions. . . ." Examiner Wai indicated that as long as computer-storage media is supported in the Specification, these amendments should overcome the § 101 rejections. Applicants indicated that computer-storage media can be found in the Specification at ¶ [0023]. Next, the § 103 rejections were discussed in regard to the AIPA used to reject dependent claims 3-4 and 15-16. Applicants' representatives disagreed that backward compatibility, as defined in the background section of the Specification, taught the elements of claims 3-4 and 15-16. Examiner Wai agreed, but indicated further searching would be required.

Rejections based on 35 U.S.C. § 101

Claims 13-16 were rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter. Claim 15 has been canceled, and as such the rejection of claim 15 is rendered moot. The Office Action states that the "claims are directed to a signal directly or indirectly by claiming a medium and the Specification recites evidence where the computer

readable medium is define as a ‘wave’ (such as a carrier wave).” Claims 13-16 have been amended herein. Claim 13, as amended, now recites “one or more computer-storage media having computer-executable instructions embodied thereon that, when executed, perform a method for...” Support in the Specification for this language can be found at ¶ [0023]. Thus, Applicants respectfully request the Examiner withdraw the 35 U.S.C. § 101 rejections of claims 13-14 and 16.

Rejections based on 35 U.S.C. § 103(a)

A.) Applicable Authority

The basic requirements of a *prima facie* case of obviousness are summarized in MPEP §2143 through §2143.03. In order “[t]o establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success [in combining the references]. Finally, the prior art reference (or references when combined) must teach or suggest all of the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art and not based on applicant’s disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991)”. See MPEP §2143. The Supreme Court in *Graham v. John Deere* counseled that an obviousness determination is made by identifying: the scope and content of the prior art; the level of ordinary skill in the prior art; the differences between the claimed invention and prior art references; and secondary considerations. *Graham v. John Deere Co.*, 383 U.S. 1 (1966). To support a finding of obviousness, the initial burden is on the Office to

apply the framework outlined in *Graham* and to provide some reason, or suggestions or motivation found either in the prior art references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the prior art reference or to combine prior art reference teachings to produce the claimed invention. *See, Application of Bergel*, 292 F. 2d 955, 956-957 (1961).

Recently, the Supreme Court elaborated, at pages 13-14 of the *KSR* opinion, that “it will be necessary for [the Office] to look at interrelated teachings of multiple [prior art references]; the effects of demands known to the design community or present in the marketplace; and the background knowledge possessed by [one of] ordinary skill in the art, all in order to determine whether there was an apparent reason to combine the known elements in the fashion claimed by the [patent application].” *KSR v. Teleflex*, 127 S. Ct. 1727 (2007). Further, in establishing a *prima facie* case of obviousness, the initial burden is placed on the Examiner. “To support the conclusion that the claimed invention is directed to obvious subject matter, either the references must expressly or impliedly suggest the claimed invention or the examiner must present a convincing line of reasoning as to why the artisan would have found the claimed invention to have been obvious in light of the teachings of the references. *Ex parte Clapp*, 227 USPQ 972, 972 (Bd. Pat. App. & Inter. 1985).” *Id.* *See also* MPEP §706.02(j) and §2142.

B.) Obviousness Rejections Based on U.S. Patent No. 5,129,084 to Kelly, Jr. et al.

Claims 1 and 13 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,129,084 to Kelly, Jr. et al (hereinafter the “Kelly reference”). As the Kelly reference fail to teach or suggest all of the limitations of each of the rejected claims, a *prima facie* case of obviousness has not been established, and Applicants therefore respectfully traverse this rejection, as hereinafter set forth.

Claim 1, as amended herein, recites, in a threaded computing environment having a plurality of contexts, each context capable of containing a queue, context settings, a context dictionary, and objects, a method for allocating the access of threads to a user interface context, the method comprising: receiving a request to access the user interface context from a first thread; determining whether the user interface context is presently being accessed by a second thread, and: if the user interface context is presently being accessed by a second thread, denying the request to access the user interface context received from the first thread; if the user interface context is not presently being accessed by a second thread, allowing the request to access the user interface context received from the first thread; maintaining thread settings associated with threads; maintaining context settings in the user interface context; and applying the context settings of the user interface context in place of the thread settings of any thread accessing the user interface context.

The Kelly reference, on the other hand, is generally directed toward an object based operating system for a multitasking computer system. *See Kelly Reference*, Abstract. The invention of the Kelly reference is also directed toward a multifaceted access control system, which supports multiple levels of visibility, allowing objects to be operated on only by processes with the object's range of visibility. *See id.* at col. 2, ll. 17-21. Mutexes, or flags, govern access to each and every container directory, and the purpose of a mutex is to ensure that only one thread accesses a particular resource at any one time. *See id.* at col. 7, ll. 26-32. Mutexes are used in the invention of the Kelly reference to synchronize access to container directories, object containers, linked lists of objects, and other types of data structures. *See id.* at col 7, ll. 40-43.

The Office Action states that "Kelly does not teach maintaining context settings in the user interface context; and applying the context settings of the user interface context in place

of the thread settings of any thread accessing the user interface context.” *Office Action*, p. 6. The Office Action cites to Applicant’s Admitted Prior Art (AAPA) in the Background of the Specification for rejecting claims 3 and 15, the elements of which are now incorporated into independent claims 1 and 13. *See id.* at p. 6. It is respectfully submitted that the Background of the Specification cited to by the Office Action does not teach or suggest allowing the request to access the user interface context received from the first thread; maintaining thread settings associated with threads; maintaining context settings in the user interface context; and applying the context settings of the user interface context in place of the thread settings of any thread accessing the user interface context.

The portion of the Background of the Specification cited to by the Office Action generally discusses backward compatibility, which is defined in the Specification as “the ability of a computer system to operate prior generations of software applications in a newer operating system.” *Specification*, ¶ [0009]. The Background of the Specification further describes backward compatibility as stating that “it is often desirable to provide for backward compatibility in an elegant fashion that maintains the stability of the new operating system while still permitting older software applications to operate.” *Id.* Contrary to this description of backward compatibility in the Specification, the elements of claim 3 and 15, now incorporated in independent claims 1 and 13, are directed to applying context setting of the user interface context *in place of* the thread settings of any thread accessing the user interface context. It is respectfully submitted that applying context settings in place of thread setting for a particular thread is not consistent with the concept of “the ability of a computer system to operate prior generations of software applications in a newer operating system.” *Id.*

Thread settings are described in the Specification as containing information, such as cultural information, that may be useful to a particular thread in processing and object within a context. *See Specification*, ¶ [0035]. Dictionaries may also be included in thread settings. *See id.* It is important to note the reason for adapting thread settings to include context settings. As stated in the Specification, “[w]hile all information necessary for operation of a thread, such as thread 510, may be in thread settings 530, method in accordance with the present invention allow information useful to threads in processing jobs to be ***context specific, rather than thread specific.***” *Id.*

Figure 6 of the Specification illustrates a thread 510 accessing context 310. *See Specification*, FIG. 6 and ¶ [0036]. While the thread is operating within this context, the thread settings 530 may actually become the context settings 316. *See, id.* These context settings are maintained within the context. *See, id.* The Specification also states that context dictionary, such as 312 in Fig. 6, and which is maintained within context 310, may also be placed within thread settings 530 while the thread is operating within the context 310. *See, id.* There are many reasons and advantages for the thread using context settings while operating within that context. For example, adapting thread setting to include the settings from the context “allows settings and dictionary information to be specified at the context level, rather than on a thread by thread basis.” *See, id.* Further, adapting thread settings “allows additional services, such as catching exceptions caused by the thread while executing within the context, to be provided.” *See Specification*, FIG. 6 and ¶ [0036].

As such, it is respectfully submitted that the Background of the Specification, specifically the section relating to backward compatibility, is not applicable to the elements of claims 3 and 15, all of which are now incorporated into independent claims 1 and 13, as

amended herein. As defined and described in the Background of the Specification, backward compatibility involves computer systems operating prior generations of software applications in a newer operating system. *See Specification*, ¶ [0009]. To the contrary, adapting thread settings to include settings from the context that the thread is operating in is not even comparable to using a prior generation of software applications, and more specifically, it is not comparable to using prior version or generation of a setting. The context setting is not a prior version, but it is a *completely different setting that replaces the thread setting for the time that the thread is in operation in the context.*

Claim 13, as amended herein, recites one or more computer-storage media having computer-executable instructions embodied thereon that, when executed, perform a method for allocating the access of threads to a user interface context in a threaded computing environment having a plurality of contexts, each context capable of containing a queue, context settings, a context dictionary, and objects, the method for allocating the access of threads to a user interface context comprising: receiving a request to access the user interface context from a first thread, wherein the user interface context comprises one or more objects; determining whether the user interface context is presently being accessed by a second thread, and: if the user interface context is presently being accessed by a second thread, denying the request to access the user interface context received from the first thread; if the user interface context is not presently being accessed by a second thread, allowing the request to access the user interface context received from the first thread; maintaining thread settings associated with threads; maintaining context settings in the user interface context; maintaining context dictionary in the user interface context, wherein the context dictionary comprises information from one or more sources; and applying the context

settings and the context dictionary of the user interface context in place of the thread settings of any thread accessing the user interface context.

For the same reasons presented above in regard to independent claim 1, as amended, it is respectfully submitted that the portion of the Background of the Specification cited to by the Office Action that purportedly teaches maintaining context settings in the user interface context; and applying the context settings of the user interface context in place of the thread settings of any thread accessing the user interface context, in fact, does not. *See Office Action*, p. 6.

The portion of the Background of the Specification cited to by the Office Action generally discusses backward compatibility, which is defined in the Specification as “the ability of a computer system to operate prior generations of software applications in a newer operating system.” *Specification*, ¶ [0009]. The Background of the Specification further describes backward compatibility as stating that “it is often desirable to provide for backward compatibility in an elegant fashion that maintains the stability of the new operating system while still permitting older software applications to operate.” *Id.* Contrary to this description of backward compatibility in the Specification, the elements of claim 3 and 15, now incorporated in independent claims 1 and 13, are directed to applying context setting of the user interface context *in place of* the thread settings of any thread accessing the user interface context. It is respectfully submitted that applying context settings in place of thread setting for a particular thread is not consistent with the concept of “the ability of a computer system to operate prior generations of software applications in a newer operating system.” *Id.*

In addition, neither the Kelly reference nor the portion of the Background of the Specification pointed to by the Office Action on p. 3 teaches or suggests maintaining context

dictionary in the user interface context, wherein the context dictionary comprises information from one or more sources. As discussed in the Specification, “[c]ontext dictionary 312 maintained within context 310 may likewise be placed within thread settings 530 while thread 510 operates within context 310.” *Specification*, ¶ [0036]. Using context settings and context dictionary as thread settings “further allows additional services, such as catching exceptions caused by thread 510 while executing within context 310, to be provided.” *Id.*

As such, it is respectfully submitted that the Kelly reference fails to teach or suggest all of the limitations of independent claims 1 and 13, as amended herein, and as such, a *prima facie* case of obviousness of claims 1 and 13 cannot be established utilizing the Kelly reference. Accordingly, Applicant respectfully requests withdrawal of the rejection of independent claims 1 and 13 under 35 U.S.C. §103(a). Independent claims 1 and 13 are believed to be in condition for allowance and such favorable action is respectfully requested.

Claims 2 and 14 are rejected under § 103(a) as being unpatentable over the Kelly reference in view of U.S. Patent No. 6,293,712 to Coutant (hereinafter the “Coutant reference”). Claims 3-4 and 15-16 are rejected under § 103(a) as being unpatentable over the Kelly reference in view of Applicant’s Admitted Prior Art (hereinafter “AAPA”). Claims 3 and 15 have been canceled herein, and as such, these rejections are rendered moot.

Each of claims 2, 4, 14, and 16 depends, either directly or indirectly, from independent claims 1 or 13 and, accordingly, it is respectfully submitted that the Kelly reference, the Coutant reference, and the AAPA, whether taken alone or in combination, fail to teach or suggest all of the limitations of these claims for at least the above-cited reasons. As such, withdrawal of the 35 U.S.C. §103(a) rejections of claims 2, 4, 14, and 16 is respectfully

requested. Each of claims 2, 4, 14, and 16 is believed to be in condition for allowance and such favorable action is respectfully requested.

CONCLUSION

For at least the reasons stated above, claims 1-2, 4, 13-14, and 16 are now in condition for allowance. Applicants respectfully request withdrawal of the pending rejections and allowance of the claims. If any issues remain that would prevent issuance of this application, the Examiner is urged to contact the undersigned – 816-474-6550 or emcfarland@shb.com (such communication via email is herein expressly granted) – to resolve the same. It is believed that no fee is due, however, the Commissioner is hereby authorized to charge any amount required to Deposit Account No. 19-2112.

Respectfully submitted,

/ELENA K. McFARLAND/

Elena K. McFarland
Reg. No. 59,320

EKM/bp
SHOOK, HARDY & BACON L.L.P.
2555 Grand Blvd.
Kansas City, MO 64108-2613
816-474-6550